# Introduction to PFC’s BorealCloud

The Pacific Forestry Centre’s Boreal Cloud infrastructure is a high performance computing (HPC) resource developed to help PFC researchers test their applications, software, algorithms and experiments before deploying their studies on larger commercial/enterprise HPC systems like Compute Canada, WestGrid, Shared Services Canada (SSC) General Purpose Science Computing, SSC ECCC Dorval Super Computing (Meterological services), or Amazon AWS.

PFC’s BorealCloud is developed on the open source technology known as OpenStack. OpenStack is considered "Infrastructure-as-a-Service" or "IaaS". It is a framework that controls large pools of compute, storage and networking resources managed through a dashboard or via the OpenStack API.

This is notably different from other HPC environments in that it does not natively support a queuing system. Users are given the flexibility to create their own server(s) and infrastructure to support their use case.

Please contact the NFIS Project Office if you have any questions on the suitability of BorealCloud for your high performance computing needs. We suggest you write [support@nfis.org](https://gcdocs.gc.ca/nrcan-rncandav/nodes/21521529/support%40nfis.org) or contact the NFIS Project Office or PFC Management for advice.

**The BorealCloud infrastructure is for UNCLASSIFIED data only**

The BorealCloud computing resource is an ongoing research project. It may go down for upgrades and maintenance with little notice to users.

# BorealCloud’s Resources

BorealCloud’s computing resources are part of the latest scientific computing resources at the Pacific Forestry Centre. PFC funded an incubator project to test the feasibility of high performance computing and to help researchers with their daily high performance computing needs.



BorealCloud is based on [OpenStack](http://www.openstack.org) (<http://www.openstack.org>). OpenStack is a group of services and hardware to support the quick and easy provisioning of virtual resources (including virtual networks, virtual servers (compute), and storage). The basic components of BorealCloud, as shown in the image above, consist of:

* [OpenStack](http://www.openstack.org) infrastructure
  + compute nodes (on which server instances will run)
  + infrastructure nodes (services that manage OpenStack)
  + storage nodes (NAS and distributed storage)
* Services the facilitate access to OpenStack:
  + Workstations
  + A "Jump" machine for remote access
  + A firewall
  + Additional NAS storage
  + User Account management and single-sign-on

For the purposes of this document, we will describe the compute, storage, and network capacity of the cluster.

**SHARING**

These resources are shared by all researchers on the BorealCloud.

Please be considerate and reduce your usage when not needed.

If necessary, please **SHELVE** your instances when not in use. **Simply shutting down instances does not free up compute resources.**

## Compute

The BorealCloud has 18 compute nodes for a total of:

* 1016 threads
* 21504 CUDA cores (10x NVidia M5000 cards, 1x NVidia P2000 card)
* 12TB RAM

When you create a server in the BorealCloud, your server will automatically be assigned to one of the compute nodes, depending on availability and the flavor selected (Flavors will be described later in this document).

To use the CUDA cores, a special flavor will need to be used. Contact the NFIS Project Office if you are interested in using CUDA cores.

## Storage

Storage is provided in the BorealCloud through a mix of traditional NAS-based storage devices and through Ceph distributed storage. This is not permanent storage space, input/output data and temporary files must be removed after processing. Data cannot be assumed to be "safe" as there is no capacity to backup or archive data.

The following is the storage availability:

* 42TB Ceph storage for OpenStack images, volumes, and VM root disk
* 25TB Ceph SSD storage for OpenStack volumes
* 96TB NAS storage available for OpenStack volumes
* 96TB NAS storage for public datasets (Sentinel 1, Sentinel 2, landsat, etc)
* 96TB NAS storage for public shared folders
* 96TB NAS storage for home directories and private project storage

Accessing this storage is described under the section ***Using the BorealCloud***and subsection ***BorealCloud storage options****.*

**IMPORTANT**

**There is no capacity to archive, permanently store, or backup data on this infrastructure.**

## Network

The BorealCloud is connected to the BC Government 1 gigabit connection to the internet. This network does NOT connect to the Government of Canada’s Natural Resources Canada network.

Network connectivity within the BorealCloud is provided by 2 sets of 2 bonded Gigabit links. Ceph distributed storage is facilitated through the use of 40Gbps infiniband networking.

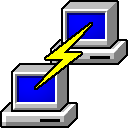
In the future, the plan is to upgrade some of the 1Gbps links to 10Gbe in order to improve I/O performance between virtual machines and to NAS storage.

# Setting up Your Computer

To connect to and work with BorealCloud systems you may have to install one or more software packages on your own computer. Although web browser-based tools may become available for accessing BorealCloud in the future, especially as Web services are developed, most users will continue to log in and work directly with remote systems with command line operations.

## Terminal client supporting ssh

The most important piece of software you will need is a terminal (client) program that supports the secure shell (SSH) protocol for network communications to remote servers.

* Linux, Unix, and Mac OS X users:
  + Use the built-in command-line SSH client. Type "ssh" at the command line.
* Microsoft Windows:
  + Will usually need to install or download a client.
  + There are two options recommended here:
    - PuTTY (<https://www.putty.org/>) [](https://www.putty.org/)
      * Does not require administrative privileges to use, if you download the putty.exe file (rather than the .msi installer).
      * Download the 64-bit putty.exe under "Alternative binary files". A direct link is here: <https://the.earth.li/~sgtatham/putty/latest/w64/putty.exe>
    - MobaXTerm ([http://mobaxterm.mobatek.net](http://mobaxterm.mobatek.net/))
      * MobaXterm has the advantage of providing a built-in graphical file transfer program and X Window display server (see below).
  + There is also an extensive list of additional SSH clients at:

<http://en.wikipedia.org/wiki/Comparison_of_SSH_clients> .

## File transfer client supporting scp and sftp

You will also need software that supports secure transfer of files between your computer and the BorealCloud machines.

* Linux, Unix, and Mac OS X users:
  + Use the built-in command-line SCP or SFTP program. Type "scp" at the command line to use it.
* Microsoft Windows:
  + pscp and psftp comes with [PuTTY](http://www.chiark.greenend.org.uk/~sgtatham/putty/)
  + [WinSCP](http://winscp.net/eng/index.php) is a free file transfer program
  + [MobaXterm](http://mobaxterm.mobatek.net/) has a graphical user interface for file transfers

## Microsoft Remote Desktop Connection for Windows

A Remote Desktop connection to a Windows virtual machine on the BorealCloud can be made either from one of the workstations in room 210 or through an SSH tunnel. This will be described in more detail below.

# Connecting to the BorealCloud

In order to connect to the BorealCloud, you will need logon credentials.

The BorealCloud is not connected to the NRCan network. There are three scenarios for which connecting to the BorealCloud is different:

1. Connecting via a workstation in Room 210
2. Connecting from a workstation in PFC on the NRCan network
3. Connecting from outside of PFC

## Connecting via a workstation in Room 210

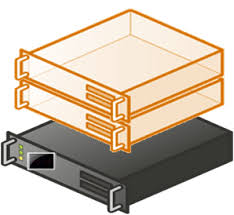
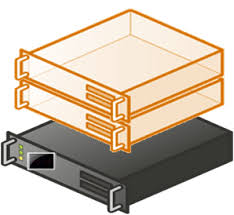
There are two workstations available in room 210 that have direct access to the BorealCloud. Simply logon to the workstation using your BorealCloud username and password.

Data can be transferred to the BorealCloud by attaching USB drives to the workstations and copying onto the BorealCloud.

## Connecting from a workstation in PFC on the NRCan network

### The Process

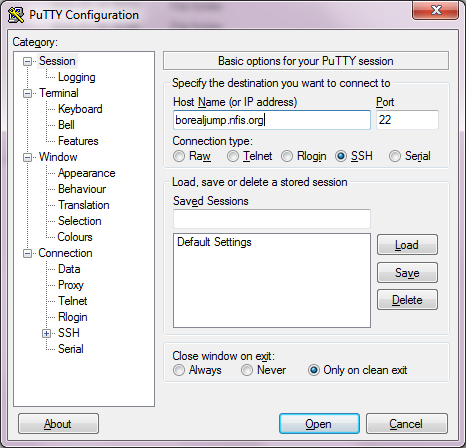
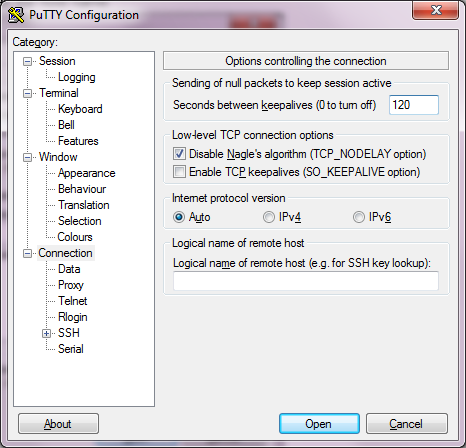
Access to the BorealCloud from your PFC workstation on the NRCan network is done via SSH:

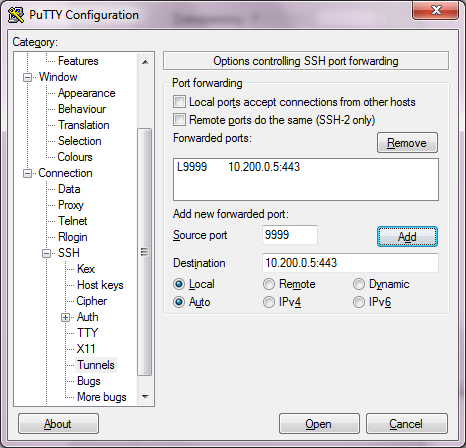
 ssh  ssh  

PFC Desktop Borealpfc.nfis.org BorealCloud (and to your VMs)

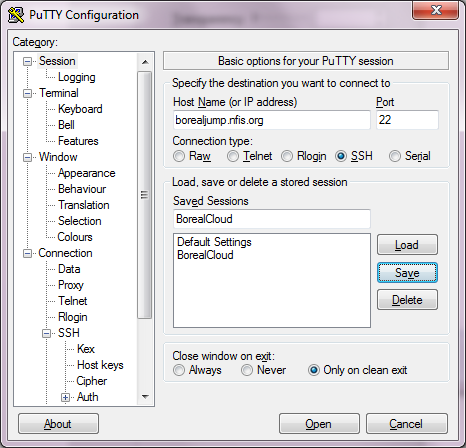
### Windows SSH Client

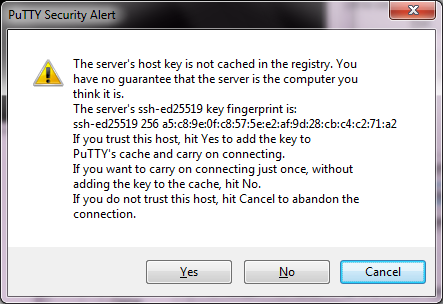
As mentioned earlier, you will need an SSH client. Putty and MobaXTerm are both free options available for Windows users. This section will describe how to use Putty to connect to the BorealCloud:

1. Double-click on the downloaded putty.exe file.
2. Enter "borealpfc.nfis.org" as the host name. **NOTE**: From inside the PFC, this has changed from “borealjump.nfis.org” or “borealcloud.nfis.org” to “borealpfc.nfis.org”.
3. Click on "Connection" and enter "120" next to "Seconds between keepalives"
4. **Optional**: Expand the "SSH" category, click on Tunnels, enter "9999" for "Source port", "10.200.0.5:443" for "Destination" and click "Add".

This will allow you to go to <https://localhost:9999/> in your web browser to visit the OpenStack web portal (while your SSH session is connected).

1. Click back to "Session", enter a name for the Saved Session (ie: BorealCloud) and click "Save".



1. Double-click on "BorealCloud" or click "Open" to connect. If this is the first time you are connecting, you will get a Security Alert like below. Click Yes. 
2. Enter your username and password. You will now be connected to the BorealCloud. If this is the first time you are connecting, you will be asked to change your password.

### Linux SSH Client

Linux, Unix, MAC, and other operating systems will often come with a command-line SSH client. You can access it using "ssh" at the command line.

1. To connect to just the command line in BorealCloud, use the following command:

ssh {{ your\_username }}@borealpfc.nfis.org

1. To connect with access to the OpenStack web portal, use the following command:

ssh -L9999:10.200.0.5:443 {{ your\_username }}@borealpfc.nfis.org

**For advanced users at the PFC**

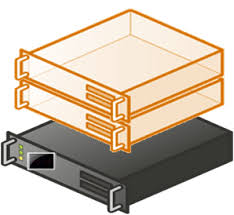
Rather than tunneling individual ports, users may consider using a Proxy setting in a Web browser to forward Web traffic over ssh. In Linux, the SSH command would be similar to the following:

ssh –D 8080 –C –N [**your\_username@borealpfc.nfis.org**](https://gcdocs.gc.ca/nrcan-rncandav/nodes/21521529/your_username%40borealjump.nfis.org) –N –V

You then need to configure your desktop’s Web browser for a manual proxy with SOCKS host: 127.0.0.1 at Port 8080 (select SOCKS v5). Remember to turn off your proxy when you are not connecting to the BorealCloud.

## Connecting from outside of PFC

Access to the BorealCloud from outside of PFC is done via SSH. However, it requires connecting through two servers (and using two sets of usernames/passwords) before reaching the BorealCloud:

 ssh Image result for firewall clipart ssh  ssh  

User’s Desktop Borealremote.nfis.org Borealjump.nfis.org BorealCloud

(and to your VMs)

**NOTE**

**Your BorealRemote username is different from your BorealCloud username. If you need access from outside of PFC, request access from the NFIS Project Office.**

To connect to the BorealCloud system from anywhere outside of the Pacific Forestry Centre you will need to remotely connect to the machine borealremote.nfis.org. The steps are the same as if you were connecting from within PFC (for Putty or from the command line) except for the following:

* Use the hostname "borealremote.nfis.org"
* Do not specify a tunnel for accessing the OpenStack web portal in the same way. It will not work in this scenario. Instead, you can use a command similar to the following:

ssh -J borealRemoteUserName@borealremote.nfis.org individualBorealCloudUserName@borealjump.nfis.org -L9999:10.200.0.5:443

A similar command could be used for tunnelling services from VMs (change the port and public IP address).

NOTE: not all versions of SSH have the “-J” parameter.

* For normal connections, use your "BorealRemote" username and password initially. You will then be prompted for your BorealCloud username. If your username has not been allowed access, you will be disconnected. Otherwise you will be prompted to enter your BorealCloud password.

# Transferring files to the BorealCloud

Files and data can be transferred in several ways:

* By using a USB drive and connecting it to a workstation in room 210. This is the fastest way.
* By using Secure Copy (SCP) from your PFC workstation to the BorealCloud.
* By pulling data from online resources (Git, Westgrid, FTP sites, etc)

## Using a USB Drive

1. Plug the USB drive into your workstation
2. Copy the files onto the USB drive
3. Disconnect and take it to room 210.
4. Login to a workstation in room 210 and plugin your USB drive.
5. Copy files from the USB drive to your home directory, public folder, or project folder. More information on the storage options is in a later section.

## Using Secure Copy (SCP)

**NOTE**

**This option is only available to users within PFC's NRCan network. For large copies, this will be extremely slow due to NRCan bandwidth limitations.**

Connecting to the Boreal Cloud using SCP is very similar to using an SSH tool. Using WinSCP or a graphical file transfer tool is straight forward. When using Linux, Unix, or a Mac, the command would be similar to:

scp myfile.txt myusername[@borealpfc.nfis.org](https://gcdocs.gc.ca/nrcan-rncandav/nodes/21521529/%7D%7D%40borealjump.nfis.org):

This will copy the file "myfile.txt" to your home directory on borealpfc.nfis.org. To copy to your project directory, specify the path like so:

scp myfile.txt myusername[@borealpfc.nfis.org:/projects/](mailto:%7d%7d@borealjump.nfis.org:/projects/)myprojectname

To copy directories, add the "-r" flag:

scp -r mydirectory myusername[@borealpfc.nfis.org:/projects/](mailto:%7d%7d@borealjump.nfis.org:/projects/)myprojectname

## Pulling from Online Resources

On the BorealCloud, you will only have access to the Linux command line. It is recommended that you use **SCP** (Secure Copy) or **SFTP** (SSH File Transfer Protocol) or **wget** (http get).

For example, copying data from an account on Westgrid, you may use a command similar to the following (to copy all .shp files to your current directory):

scp [user1@orciunus.westgrid:/home/user1/\*.shp](mailto:user1@orciunus.westgrid:/home/user1/*.shp) .

If you needed to download data from the NFIS Forest Change Portal to the BorealCloud you would use the following Linux command wget:

wget https://opendata.nfis.org/downloads/forest\_change/C2C\_Change\_Type.zip

# Logging-in to OpenStack once on the BorealCloud

## Using BorealCloud command line

After you login to the borealcloud remotely, you will encounter the command line:

[myusername@jump ~]$

Now:

1. Look for a subdirectory called *quickstart*:

[myusername@jump ~]$ ls quickstart/

create-server.yml openrc.sh readme.txt

There are three files in the quickstart/ directory:

* readme.txt
  + This describes how to create your first virtual machine
* openrc.sh
  + This is a script that allows you to authenticate with OpenStack in order to create your first virtual machine
* create-server.yml
  + This is an example “ansible” script which provides an alternate way to create a virtual machine (instead of using “openstack” commands).

1. View the readme.txt file under *quickstart/* for a quick introduction to OpenStack commands:

[myusername@jump ~]$ less quickstart/readme.txt

1. When you are ready, login to OpenStack like so (you will be prompted for your password):

$ source quickstart/openrc.sh

Your password is the same as your BorealCloud password.

In addition to the OpenStack commands outlined in the quickstart/readme.txt file, there is an overview of OpenStack commands later in this document.

### Commands that run when you login

For experienced users of Linux, environment variables, aliases, and commands are sometimes added to files .bash\_rc or .bash\_profile in order to have them run when the user logs-in.

**This will not work with the BorealCloud.**

Some files are regularly overwritten by an automatic process – in order for administrators to be able to update and improve the user's experience. This includes files such as .bashrc and .bash\_profile.

In order to have commands run or environment variables set at login time, please create and add them to the following file instead:

* **.bashrc\_local**

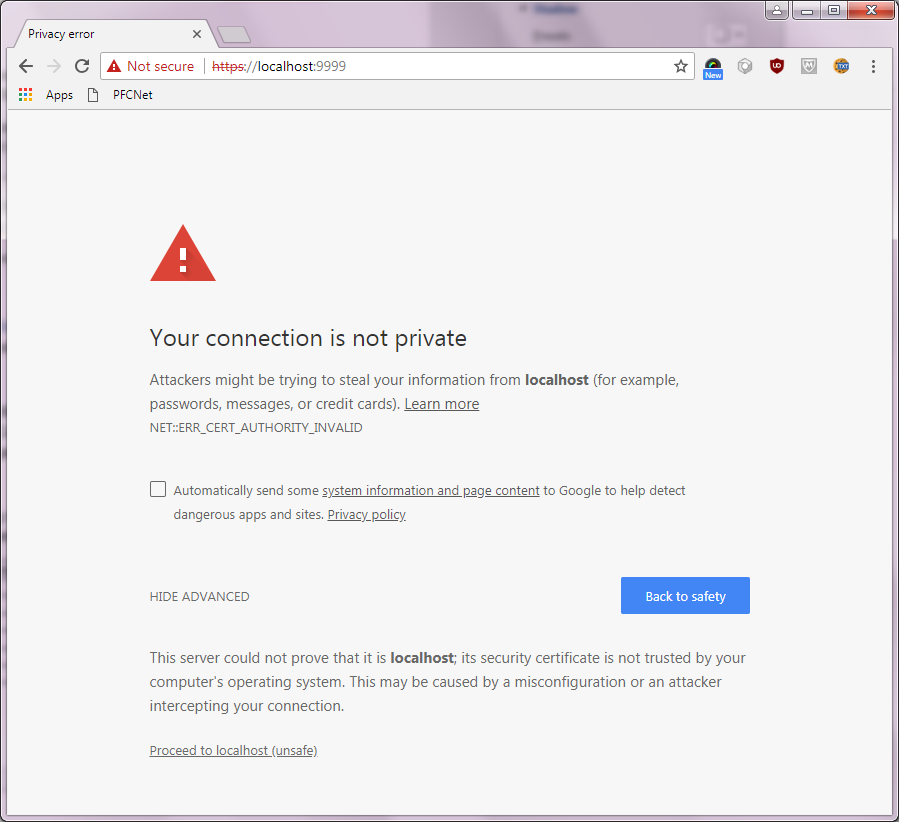
This file will NOT be overwritten and will be executed at login.

## Using the OpenStack web portal

### Via a workstation in room 210

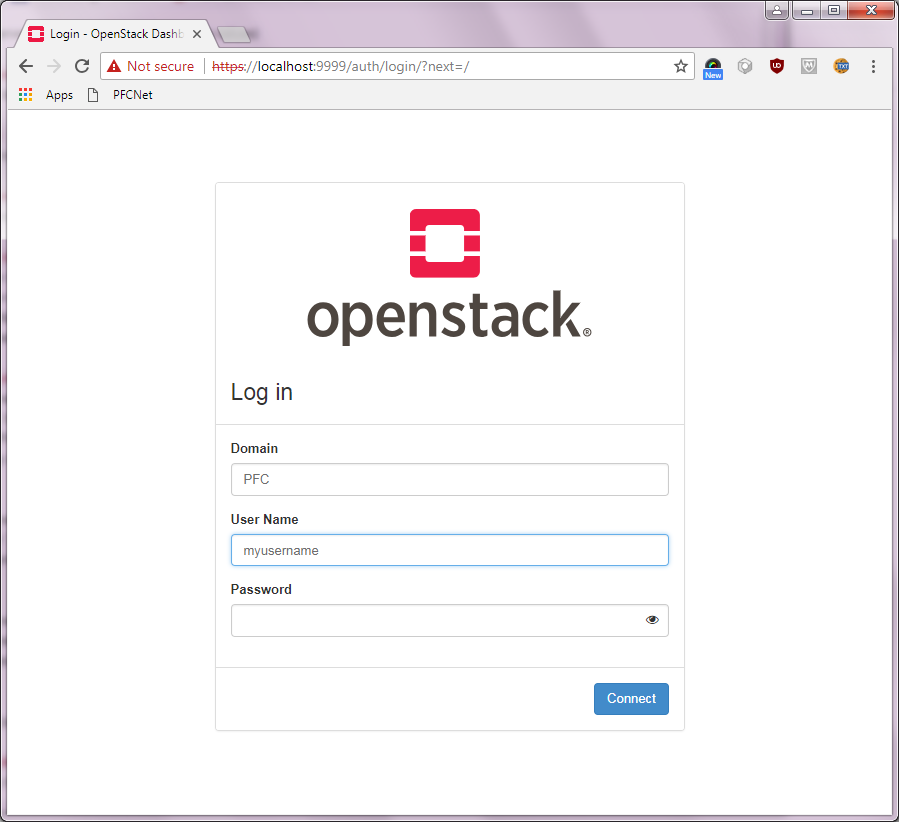
1. Login to the workstation in room 210 using your BorealCloud username and password.
2. Open a web browser and visit URL <https://10.200.0.5/> in a web browser.
3. Follow the steps under "Via a Port Tunnel" starting at step 3.

### Via a Port Tunnel

1. Configure a port tunnel on port 9999, as described under the *Connecting to the BorealCloud* section.
2. Visit the URL <https://localhost:9999/> in a web browser.
3. You will get a security warning because the Certificate used for secure communications is not verified by a third party (and it was signed against a different address than "localhost:9999"). The warning, as shown in Google Chrome, is displayed below. Click on "Advanced" and "Proceed to localhost". Other browsers will display the warning differently and will have a different way to bypass it. 
4. You should now see the OpenStack logon page (it may take some time to load initially). Login using:

**Domain**: PFC

**Username**: *myBorealCloudUserName (the same as your BorealCloud SSH username)*

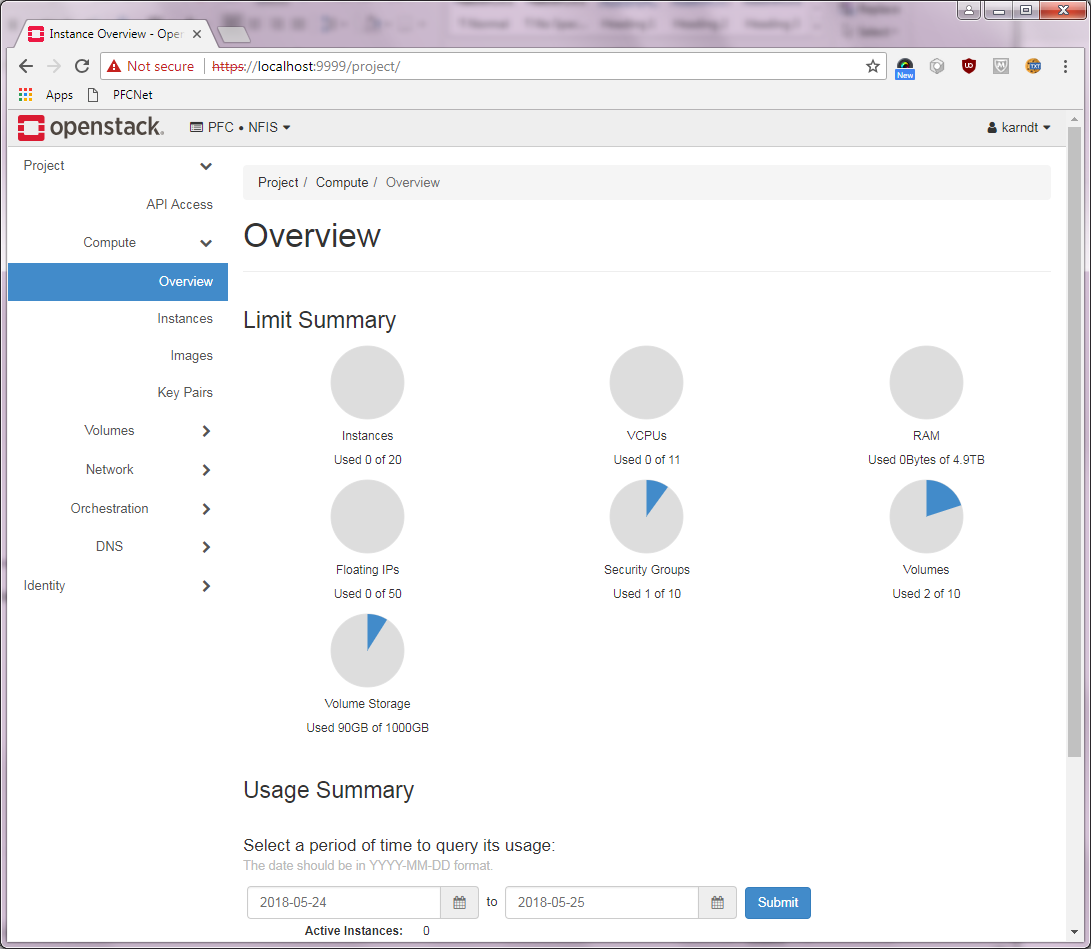
**Password**: *myBorealCloudPassword (the same as your BorealCloud SSH password)* 

1. You will now see the OpenStack overview page.

On the left is the menu. The most important menu items for new users are:

* **Instances**: where you can manage your servers
* **Images**: where you can manage the Operating System and software that your servers will run
* **Key Pairs**: where you can create credentials to access your servers
* **Volumes**: where you can create new "hard drives" to add space to your servers

More information on these concepts can be seen later in this document.



1. Navigating the web portal will be left up to the user to explore. More information on the concepts will be presented later in this document. More information can also been seen on the OpenStack website: <https://docs.openstack.org/horizon/latest/user/index.html>

# Using the BorealCloud

When you wish to create a new server, you need several things:

* Storage (where do I store stuff)
* Compute (how big does my server need to be, how many do I need)
* Operating System (what Operating System/software image do I want to start with)
* Network (how do I connect a server to the network)
* Access (how can I login to my server)

This section describes each of these components. The OpenStack command-line interface that will make use of these components is described in a later section.

## Storage

Once you have logged-on to the Boreal Cloud or one of the workstations in room 210, the following storage options are available:

* **/export/home/myusername**
  + This is your "home" directory.
  + This is the directory you are in when you first login.
  + Represented by the tilde "~" when shown in the command prompt ([myusername@jump ~]$)
  + This storage is not available directly from within a server that you create. You can use "SCP" to copy files from here to a server that you create.
* **/projects/projectname**
  + This is your "project" directory
  + This is shared storage that is available to only members of your project
  + This storage is not available directly from within a server that you create. You can use "SCP" to copy files from here to a server that you create.
* **/public**
  + This is a "public" free-for-all directory
  + Everyone has read/write to this directory
  + Create a subdirectory for your project here (/public/projectname) to keep your data separate from other projects. This is not a guarantee of privacy.
  + This folder can be mounted on linux using a command like:
    - mount -t nfs 10.20.0.6:/Public /mnt
  + This folder can be seen in windows by going to:
    - \\10.20.0.6\Public
* **/data**
  + This contains read-only publicly-available datasets
  + This is for Datasets that are of general interest
  + Sentinel1 and Sentinel2 are examples
  + This folder can be mounted on linux using a command like:
    - mount -t nfs 10.20.0.7:/publicdata /mnt
  + This folder can be seen in windows by going to:
    - [\\10.20.0.6\publicdata](file:///\\10.20.0.6\publicdata)
* **OpenStack Volumes**
  + Servers that you create will generally only come with 20GB of root disk
  + You can create "volumes" in OpenStack to add additional storage to your servers.
  + When you create an empty volume and attach it to a server, it appears within the server as an unformatted hard drive. You can format it using Windows or Linux utilities.
  + You can also create servers with their root disk on a volume. This allows your root disk to be of any size.
  + Volumes can be moved from one server to another – like taking a hard drive out of one computer and putting it in another.
  + Creating volumes on the command line will be described later in this document.
  + Creating volumes in the web interface is done under the "Volumes" menu item.

When you upload data to the BorealCloud, you have the choice to store it under your home directory, project directory, or the public directory.

## Compute (ie: Flavors)

Server instances can be created with varying resources to meet scientific and computational needs. The following virtual machine sizes and allocations are available to researchers. These are sized in order to make optimum use of the underlying hardware.

Custom configurations are not recommended as they may leave parts of the cluster underutilized.

To enable Cuda-core processing, additional flavors can be made available. Contact the NFIS Project Office for details.

**NOTE**

**The # Instances Available column assumes that the underlying hardware is dedicated to instances running that flavor. In practice, the actual number of instances that can be created using the flavor will be less. This number also represents the total number of possible instances for all OpenStack users.**

**If you are unable to create the desired number of servers using a particular instance, try a different size (particularly a smaller sized flavor) or a different group (ie: gp2 vs mm1) as different groups will run on different sets of hardware.**

**UPDATE (January, 2020):**

**The "gp1" & “gp2” instances have been removed as the nodes they served have been upgraded with additional RAM. The “mm2” instances have now been made available to all.**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Flavor | Cores | Memory (MB) | Root Disk (GB) | # Instances available | Comments |
| ~~gp1.tiny~~ | ~~1~~ | ~~512~~ | ~~20~~ | ~~224~~ | No longer available. |
| ~~gp1.small~~ | ~~2~~ | ~~1024~~ | ~~20~~ | ~~112~~ | No longer available. |
| ~~gp1.medium~~ | ~~4~~ | ~~2048~~ | ~~20~~ | ~~56~~ | No longer available. |
| ~~gp1.large~~ | ~~8~~ | ~~4096~~ | ~~20~~ | ~~28~~ | No longer available. |
| ~~gp1.xlarge~~ | ~~16~~ | ~~8192~~ | ~~20~~ | ~~14~~ | No longer available. |
| ~~gp1.2xlarge~~ | ~~28~~ | ~~14336~~ | ~~20~~ | ~~8~~ | No longer available. |
| ~~gp1.4xlarge~~ | ~~56~~ | ~~28672~~ | ~~20~~ | ~~4~~ | No longer available. |
| ~~gp2.tiny~~ | ~~1~~ | ~~1024~~ | ~~20~~ | ~~1120~~ | No longer available. |
| ~~gp2.small~~ | ~~2~~ | ~~2048~~ | ~~20~~ | ~~560~~ | No longer available. |
| ~~gp2.medium~~ | ~~4~~ | ~~4096~~ | ~~20~~ | ~~280~~ | No longer available. |
| ~~gp2.large~~ | ~~8~~ | ~~8192~~ | ~~20~~ | ~~140~~ | No longer available. |
| ~~gp2.xlarge~~ | ~~16~~ | ~~16384~~ | ~~20~~ | ~~70~~ | No longer available. |
| ~~gp2.2xlarge~~ | ~~28~~ | ~~28672~~ | ~~20~~ | ~~40~~ | No longer available. |
| ~~gp2.4xlarge~~ | ~~56~~ | ~~53248~~ | ~~20~~ | ~~22~~ | No longer available. |
| ~~gp2e.4xlarge~~ | ~~56~~ | ~~117760~~ | ~~20~~ | ~~10~~ | No longer available. |
| mm1.small | 1 | 4506 | 20 | 1,344 |  |
| mm1.medium | 2 | 9011 | 20 | 672 |  |
| mm1.large | 4 | 18022 | 20 | 336 |  |
| mm1.xlarge | 8 | 36045 | 20 | 168 |  |
| mm1.4xlarge | 16 | 72090 | 20 | 84 |  |
| mm1.8xlarge | 28 | 126157 | 20 | 48 |  |
| mm1.16xlarge | 56 | 252314 | 20 | 24 |  |
| mm2.xlarge | 4 | 36864 | 20 | 140 |  |
| mm2.2xlarge | 8 | 73728 | 20 | 70 |  |
| mm2.4xlarge | 16 | 147456 | 20 | 35 |  |
| mm2.8xlarge | 28 | 258048 | 20 | 20 |  |
| mm2.16xlarge | 56 | 500000 | 20 | 10 |  |
| mm2e.16xlarge | 56 | 1013760 | 20 | 5 |  |
| x1.xlarge | 4 | 63488 | 20 | 32 | Avail. by request only |
| x1.2xlarge | 8 | 126976 | 20 | 16 | Avail. by request only |
| x1.4xlarge | 16 | 253952 | 20 | 8 | Avail. by request only |
| x1.8xlarge | 32 | 507904 | 20 | 4 | Avail. by request only |
| x1.16xlarge | 64 | 1015808 | 20 | 2 | Avail. by request only |
| x1e.16xlarge | 64 | 2043904 | 20 | 1 | Avail. by request only |

You can view the flavors that are available to you using the OpenStack command-line utility.

### Need more than 20GB of disk space?

If you need more than 20GB of space, create a volume and attach it to your server.

Alternatively, if your image is larger than 20GB or you require more than 20GB of root disk, please create a volume-backed instance.

For this, there are 3 options:

1. The easiest way is to use an ansible script. Copy the one in quickstart/create-server.yml and change/uncomment the following:

image: { your image }

flavor: { your flavor }

boot\_from\_volume: true

volume\_size: 50

terminate\_volume: true

"volume\_size" is in GB. "terminate\_volume" will ensure the volume is also deleted when the virtual machine is deleted. You may have to adjust other parameters as well (ie: it assumes your key name is the same as your username).

1. You can create a volume-backed instance with the OpenStack command-line. There is an extra step, however. There are instructions here:

<https://docs.openstack.org/nova/pike/user/launch-instance-from-volume.html>

1. Use the OpenStack web portal to create your instance. Select “Yes” to boot from a volume and change the volume size to suit your needs.

## Images

Images are used to populate a volume or to boot a server instance.

Preconfigured Operating systems

Several images have been made available on the system. This list may not be exhaustive.

|  |  |  |  |
| --- | --- | --- | --- |
| Operating System / Image | Version | Name | Comments |
| Centos | 7 | Centos-7-x86\_64-GenericCloud-1901 | No licence necessary |
| Ubuntu | 16.04 | Ubuntu-16.04 | No licence necessary |
| Ubuntu | 18.04 | ubuntu-18.04-server-cloudimg-amd64 | No licence necessary |
| CirrOS | 0.3.5 |  | No licence necessary |
| GRDI / Linux | N/A |  | Genomics image maintained by AAFC for the GRDI initiative |

Windows images can be uploaded and used by individual projects. It requires a Windows licence for each instance for long term operations/research. See Microsoft licencing.

You can view the images that are available to you using the OpenStack command-line utility or by clicking on the "Images" menu item in the OpenStack web portal.

### Snapshots

You can also create custom images in addition to existing images by creating *snapshots.* You are encouraged to follow a development cycle similar to the following:

1. Create a server using a base image (CentOS, Ubuntu, etc)
2. Customize the server by configuring it, installing software, etc.
3. Create a snapshot of your customized server.
4. Create new servers based on your new snapshot image.
5. The new servers will contain all your customizations and software.

## Network

Virtual networks can be created, modified, and destroyed in OpenStack. Creating additional networks can be useful in order to ensure isolation from other projects or for isolation within a project. Servers, when created, must be attached to a network in order to access the internet, access storage, or be accessible via SSH.

### BorealCloud default networks

In the BorealCloud you may have access to 3 networks that have already been created for you:

* {{ Project Name }}
  + This is a network that corresponds to the name of your project (ie: radar or spades). It is a private network for the project. It is not visible from BorealCloud, from outside OpenStack, or from users in other projects.
  + Only some projects have had this network already created. However, the plan is to have all projects use their own network.
* Private
  + Contrary to its name, any user of OpenStack on the BorealCloud can create a server instance on this network. It is thus more of a "shared" network. However, it is not visible from BorealCloud or from anywhere outside of OpenStack.
  + The use of this network is now being discouraged if the project has its own network.
* External
  + This is the 10.20.0.x network that used by the BorealCloud jump server, the workstations in room 210, and other devices that are "external" to OpenStack.
  + OpenStack servers can have a presence on this network if they are allocated a "Floating IP address".

Most users, when creating server, will want to connect it to their "Project Name" network (ie: spades or radar, etc). Use of the "private" network is being discouraged if your project has its own network. A server should never be attached directly to the External network.

If SSH or RDP (Windows Remote Desktop) access to the server is required, a Floating IP address can be assigned to the server to allow it to have a 10.20.0.x address. That makes it "visible" to computers on the 10.20.0.x network (including you on the BorealCloud jump server or the workstations in room 210).

### Why would you ever have a server that is inaccessible?

There are a finite number of Floating IP addresses on the 10.20.0.x network. Thus it is prudent, where possible, to avoid using Floating IP addresses. Here are a couple scenarios where you would not need a floating IP address for every server:

* You have a "control" server and many "slave" servers. The "control" server creates new "slave" servers as needed and destroys them when not needed and specifies the location for input and output data for the slaves. You may have a floating IP address on the "control" server in order to login and configure it. However, the "slave" servers can communicate with the "control" server via the Private network.
* You create servers based on a custom image. When you create the server, you can specify "cloud-init" data that can specify a script to automatically run in the server. The server configures itself based on information provided in the cloud-init data or by fetching configuration information from a web service or git repository. The server runs its script fetching input and writing output according to its configuration. You need not login to the server and thus it need not have a floating IP address.

## Access (ie: KeyPairs)

When you create a server from some of the basic images (like CentOS & Ubuntu), you are unable to login to the server using a username and password. You are, instead, required to create and use a *keypair*. A keypair is a set of private and public keys that you create to help identify yourself. For absolute simplicity, think of the private key as though it were a physical house key. Think of the public key as your house's deadbolt. You can prove that you are you if you can insert your key into your house's deadbolt and open the door. The same is true with a private/public key pair. If you hold the private key it will unlock access to any of the deadbolts that you have attached to servers you create.

OpenStack will keep the public key (the deadbolt) in order to attach it to the servers you create. You keep the private key (the house key) in a file with extension .pem. This file you will likely want to store in your BorealCloud home directory.

When you create an Ubuntu or CentOS server, and specify a keypair, you can logon to the server using SSH by specifying your .pem file like so:

ssh -i mykeypair.pem ubuntu@10.20.0.x

Information on how to create a keypair using the OpenStack command line is described later in this document.

# Using the OpenStack command-line interface in the BorealCloud

As described earlier, you are encouraged to view the readme.txt under the subdirectory *quickstart/* in your home directory. Review earlier sections [Connecting to the BorealCloud](#_Connecting_to_the) and [Logging-in to OpenStack once on the BorealCloud](#_Logging-in_to_OpenStack) prior to this section.

For more detailed information on the “openstack” commands, see: <https://docs.openstack.org/python-openstackclient/latest/cli/command-list.html>

This section will present an overview of basic OpenStack command-line commands. There is some overlap with the commands you have already seen in the readme.txt file.

## How to list available “flavors” or sizes of Virtual Machines

openstack flavor list

## How to list available “key-pairs”. These are used to login to your VM after creation.

openstack keypair list

## How to create a new “key-pair” and adjust permissions to 0600.

openstack keypair create mynewkeypair > mynewkeypair.pem

chmod 0600 mynewkeypair.pem

Make sure that mynewkeypair.pem actually contains a value. If there was an error running the openstack keypair create command, mynewkeypair.pem might be empty.

## How to list available images (pre-configured operating systems & software).

openstack image list

## How to create a new Virtual Machine

openstack server create \

--image CentOS7 \

--key-name mynewkeypair \

--nic net-id=Private \

--flavor gp1.small \

my\_server\_name

This will create a new server using your keypair called "mynewkeypair", connect it to the "Private" network, use the "CentOS7" image, use the flavor "gp1.small", and name the new server "my\_server\_name".

## How to list available Floating IP addresses

openstack floating ip list

A floating IP is available if “Fixed IP” and “Port” have the value “None”. If all the floating IPs have been assigned (or there are no IPs in the list), create a new one in the next step.

## Create a new “floating” IP from the pool (to make it available for use by a VM)

openstack floating ip create External

## Assign a floating IP to your VM

openstack server add floating ip my\_server\_name <Available Floating IP>

## How to access your VM

ssh -i mynewkeypair.pem centos@<floating ip assigned to VM>

Note that different operating systems have different default usernames for logging-in. Centos uses “centos”, Ubuntu uses “ubuntu”, and Cirros uses “cirros”. Adjust appropriately.

## How to list all VMs in your project

openstack server list

## How to start a stopped VM

openstack server start myserver

## How to stop a running VM

openstack server stop myserver

## How to shelve a VM

openstack server shelve myserver

## How to unshelve a VM

openstack server unshelve myserver

## How to show information on a VM

openstack server show myserver

## How to list all volumes in your project

openstack volume list

## How to create a new, basic, volume (size is in gigabytes)

volume create --size 10 mynewvolume

## How to add a volume to an existing VM

openstack server add volume myserver mynewvolume

Now you can connect to your server and partition & format the volume. In CentOS, it will show up as a /dev/vd?? device.

## How to remove a volume from an existing VM

openstack server remove volume myserver mynewvolume

## How to delete an existing volume (and all its data)

openstack volume delete mynewvolume

## How to create a snapshot of an existing server

openstack server image create --name myusername\_img myusername\_vm

This will create a new "image" based on your existing server. It allows you to create a server based on CentOS or Ubuntu, add software, configure the server, and otherwise customize it. Then take a snapshot to create a new image. Now you can create new servers by specifying the name of your snapshot as the image.

## View the status of an existing image (or snapshot)

openstack image show myusername\_img

# Using the OpenStack web portal in the BorealCloud

In order to login to the OpenStack web portal, review the section [Logging-in to OpenStack once on the BorealCloud](#_Logging-in_to_OpenStack).

Additional information on using the web portal is not currently provided in this document. Most of the OpenStack components described in earlier sections are clearly labelled in the OpenStack menu.

You are encouraged to explore the web portal and visit the following website for more information on using the OpenStack web portal.

<https://docs.openstack.org/horizon/latest/user/index.html>

# Using a Programming Language Library

OpenStack can be managed using a variety of different programming languages. Programming these is beyond the scope of this document. For more information, see: <https://wiki.openstack.org/wiki/SDKs>

# Using a Provisioning or Script Tool

Provisioning or scripting tools are handy for creating repeatable, self-documenting, configuration scripts. Many of the popular provisioning tools include modules or connectors for managing OpenStack.

As described in the readme.txt, an example “Ansible” script called “create-server.yml” has been provided under the “quickstart” directory in your home directory on the BorealCloud. For more information on Ansible, see <https://www.ansible.com/>.

In order to run the ansible script, you need to authenticate. This is the same process as was done for running OpenStack commands. If you have already authenticated for running OpenStack commands, you need not authenticate again.

To authenticate, run “source quickstart/openrc.sh” from your home directory:

source quickstart/openrc.sh

Please enter your OpenStack Password for project nfis as user <yourusername>:

Enter your password as prompted. The password is the same as what you have used to logon to the BorealCloud jump server.

Now, enter the following command:

ansible-playbook quickstart/create-server.yml

You will be prompted for the name of the virtual machine you would like to create. After entering your name, and waiting for up to a minute, your virtual machine should be created. The last “task” should describe how to connect to your virtual machine.